

Please amend the present application as follows:

**Claims**

The following is a copy of Applicant's claims that identifies language being added with underlining ("\_\_\_\_") and language being deleted with strikethrough ("———") or brackets ("[[ ]]"), as is applicable:

1. (Currently amended) A method for controlling program installation on a computing device, the method comprising:

an installer program that executes on the computing device determining the configuration of an existing program that executes on the computing device;

the installer program that executes on the computing device determining the configuration of a new program that is to be installed on the computing device to replace the existing program from information stored within a software package that comprises the new program;

the installer program that executes on the computing device determining whether installation of the new program is authorized through comparison of the configurations of the new and existing programs; and

the installer program that executes on the computing device preventing installation of the new program if installation is not authorized.

2. (Original) The method of claim 1, wherein determining the configuration of an existing program comprises determining at least one of a program type and version, and determining the configuration of a new program comprises determining at least one of a program type and version.

3. (Original) The method of claim 1, wherein determining the configuration of an existing program comprises determining the configuration of a program that is embedded in solid-state memory of the computing device.

4. (Original) The method of claim 1, wherein determining the configuration of an existing program comprises determining the configuration of an operating system that is embedded in re-writable, solid-state memory of a terminal computer.

5. (Original) The method of claim 1, wherein determining the configuration of an existing program comprises reading configuration information stored in a management interface of the computing device.

6. (Original) The method of claim 1, wherein determining the configuration of an existing program comprises reading a program type and version from an original equipment manufacturer (OEM) string of a desktop management interface (DMI) of the computing device.

7. (Original) The method of claim 1, wherein determining the configuration of a new program comprises reading configuration information from a header associated with the new program.

8. (Original) The method of claim 1, wherein determining whether installation of the new program is authorized comprises comparing the existing program and the new program to determine whether they are of the same type.

9. (Original) The method of claim 1, wherein determining whether installation of the new program is authorized further comprises comparing version information for the existing program and the new program.

10-14. (Canceled)

15. (Currently amended) A ~~system stored on a~~ computer-readable medium that stores a installer program, the ~~system~~ installer program comprising:

logic configured to execute on a user computer and determine on the user computer the type and version of an existing operating system embedded in memory of ~~a computing device~~ the user computer;

logic configured to execute on the user computer and determine on the user computer the type and version of a new operating system that has been downloaded to the computing device user computer; and

logic configured to execute on the user computer and compare on the user computer the types and versions of the new and existing operating systems;

logic configured execute on the user computer and to determine on the user computer whether installation of the new program operating system is authorized based upon the comparison; and

logic configured to execute on the user computer and prevent installation of the new operating system on the user computer if installation is not authorized.

16. (Original) The system of claim 15, wherein the logic configured to determine the type and version of an existing operating system comprises logic configured to read configuration information stored in a management interface of the computing device.

17. (Original) The system of claim 15, wherein the logic configured to determine the type and version of a new operating system comprises logic configured to read configuration information from a header associated with the new operating system.

18. (Original) The system of claim 15, wherein the logic configured to determine whether installation of the new operating system is authorized comprises logic configured to compare the type of the existing operating system with the type of the new operating system.

19. (Original) The system of claim 15, further comprising logic configured to install the new operating system and replace the existing operating system when installation is authorized.

20. (Currently amended) A computing device, comprising:

a processor; and

memory comprising an existing operating system, and a management interface that comprises configuration information that describes the type and version of the existing operating system, ~~the configuration information being accessible to an installer program that is configured to install new versions of the operating system~~ and an installer program configured to (i) determine the configuration of the existing operating system from the configuration information of the management interface, (ii) determine the configuration of a new operating system that is to replace the existing operating system from information stored within a software package that contains the new operating system, (iii) determine whether installation of the new operating system is authorized through comparison of the configurations of the new and existing operating systems, and (iv) prevent installation of the new operating system if installation is not authorized.

21. (Currently amended) The device of claim 20, wherein the memory comprises re-writable, solid-state memory and wherein the existing operating system is embedded within the solid-state memory.

22. (Original) The device of claim 20, wherein the management interface comprises a desktop management interface (DMI) and the configuration information is stored in a original equipment manufacturer (OEM) string contained within the DMI.

23. (Canceled)

24. (Original) The device of claim 20, wherein the computing device is a terminal computer that does not comprise a hard drive.

25. (New) The method of claim 1, further comprising:

the computing device receiving the software package from a software source via a network, wherein the software package contains the installer program;

the computing device identifying the installer program contained in the software package; and

the computing device installing the installer program on the computing device prior to determining whether installation of the new program is authorized.

26. (New) The computer-readable medium of claim 15, wherein the installer program comprises part of a software package that is configured for download to the user computer, the software package further including the new operating system such that the installer program will be run on the user computer when the new operating system is downloaded to the user computer and a user attempts to install the new operating system on the user computer.

27. (New) The computing device of claim 20, wherein the installer program comprises part of the software package such that the installer program will be run on the user computer when the new operating system is downloaded to the user computer from a software source and a user attempts to install the new operating system on the computing device.